

# LaTeX, beamer, tikz und Co.

## 2. Grobstruktur von Dokumenten

Thomas Worsch

Fakultät für Informatik  
Karlsruher Institut für Technologie

Wintersemester 2017/2018

# Überblick

## Allgemeines

Prinzipieller Aufbau einer  $\LaTeX$ -Datei

„Feine Grobstruktur“ von  $\LaTeX$ -Dokumenten

Leerzeichen, Leerzeilen, Kommentare

# Dokumente

## Inhalt, Struktur und Form

- ▶ *Inhalt* des Textes
- ▶ *Struktur* des Textes
- ▶ *Erscheinungsbild, Form*

# Dokumente

## Inhalt, Struktur und Form

- ▶ *Inhalt* des Textes
- ▶ *Struktur* des Textes
- ▶ *Erscheinungsbild, Form*

andere Form:

- ▶ INHALT des Textes
- ▶ STRUKTUR des Textes
- ▶ ERSCHEINUNGSBILD, FORM

# Dokumente

## Inhalt, Struktur und Form

- ▶ *Inhalt* des Textes
- ▶ *Struktur* des Textes
- ▶ *Erscheinungsbild, Form*

andere Struktur:

*Inhalt* des Textes, *Struktur* des Textes und  
*Erscheinungsbild, Form*

# Dokumente

## Inhalt, Struktur und Form

- ▶ *Inhalt* des Textes
- ▶ *Struktur* des Textes
- ▶ *Erscheinungsbild, Form*

anderer Inhalt:

- ▶ *Balaenoptera musculus* (Blauwal),
- ▶ *Mesoplodon carlhubbsi* (Hubbs-Schnabelwal) und
- ▶ *Physeter macrocephalus* (Pottwal).

# Wozu T<sub>E</sub>X und Co.?

## Programme

- ▶ wie zum Beispiel *etex*, *pdftex*, *xetex*, *luatex*, *latex*, *pdflatex*, *xelatex*, *lualatex*, ...

## transformieren

- ▶ *Eingabetext* mit
  - ▶ Inhalt
  - ▶ *logischen Textauszeichnungen* (Beschreibung Struktur)
  - ▶ *visuellen Textauszeichnungen* (Beschreibung äußerer Form)

## in

- ▶ Ausgabe-*Dokument* mit *gesetztem Text*
  - ▶ dessen Visualisierung der Eingabe «entspricht»
  - ▶ üblicherweise *pdf* (früher *dvi*)

## Wozu T<sub>E</sub>X und Co. ? (2)

Transformation von Eingabetext in Ausgabedokument ergänzt  
(unter Umständen)

- ▶ «kleine Informationen»
  - ▶ z. B. automatische Nummerierung von Kapiteln
- ▶ «größere Inhalte»
  - ▶ z. B. Inhaltsverzeichnis
- ▶ farbliche, typografische, ... Hinweise für  
Visualisierung logischer Struktur
  - ▶ z. B. blaue Dreiecke für Aufzählungspunkte



# T<sub>E</sub>X: Engines und Formate

## Programme

- ▶ wie zum Beispiel *etex*, ...

## beinhalten

- ▶ eine *Engine*
- ▶ die *primitive Kommandos* versteht
  - ▶ einschließlich eines Kommandos,  
um zusätzlich neue Kommandos zu vereinbaren

## und laden beim Start üblicherweise

- ▶ ein *Format* (Definitionen neuer Kommandos)
  - ▶ z. B. für manche logischen Auszeichnungen
  - ▶ (schnelles Laden; eigene Formate machbar)

# T<sub>E</sub>X: Engines, Formate, Ausgabeformen

Prog. name	Engine	Format	Ausgabe
<code>tex</code>	<code>tex</code>	plain T <sub>E</sub> X	<code>dvi</code>
<code>etex</code>	<code>pdftex</code>	plain T <sub>E</sub> X	<code>dvi</code>
<code>latex</code>	<code>pdftex</code>	L <sup>A</sup> T <sub>E</sub> X	<code>dvi</code>
<code>pdftex</code>	<code>pdftex</code>	plain T <sub>E</sub> X	<code>pdf</code>
<code>luatex</code>	<code>luatex</code>	plain T <sub>E</sub> X	<code>pdf</code>
<code>xetex</code>	<code>xetex</code>	plain T <sub>E</sub> X	<code>pdf</code>
<code>pdflatex</code>	<code>pdftex</code>	L <sup>A</sup> T <sub>E</sub> X	<code>pdf</code>
<code>lualatex</code>	<code>luatex</code>	L <sup>A</sup> T <sub>E</sub> X	<code>pdf</code>
<code>xelatex</code>	<code>xetex</code>	L <sup>A</sup> T <sub>E</sub> X	<code>pdf</code>

# T<sub>E</sub>X: Engines, Formate, Ausgabeformen

Prog. name	Engine	Format	Ausgabe
<code>tex</code>	<code>tex</code>	plain T <sub>E</sub> X	<code>dvi</code>
<code>etex</code>	<code>pdftex</code>	plain T <sub>E</sub> X	<code>dvi</code>
<code>latex</code>	<code>pdftex</code>	L <sup>A</sup> T <sub>E</sub> X	<code>dvi</code>
<code>pdftex</code>	<code>pdftex</code>	plain T <sub>E</sub> X	<code>pdf</code>
<code>luatex</code>	<code>luatex</code>	plain T <sub>E</sub> X	<code>pdf</code>
<code>xetex</code>	<code>xetex</code>	plain T <sub>E</sub> X	<code>pdf</code>
<code>pdflatex</code>	<code>pdftex</code>	L <sup>A</sup> T <sub>E</sub> X	<code>pdf</code>
<code>lualatex</code>	<code>luatex</code>	L <sup>A</sup> T <sub>E</sub> X	<code>pdf</code>
<code>xelatex</code>	<code>xetex</code>	L <sup>A</sup> T <sub>E</sub> X	<code>pdf</code>

# Überblick

Allgemeines

Prinzipieller Aufbau einer  $\LaTeX$ -Datei

„Feine Grobstruktur“ von  $\LaTeX$ -Dokumenten

Leerzeichen, Leerzeilen, Kommentare

# Grobstruktur

```
\documentclass{\langle doc type \rangle}
```

```
\langle preamble \rangle
```

```
\begin{document}
```

```
\langle doc body \rangle
```

```
\end{document}
```

# Grobstruktur

```
\documentclass{<doc type>}
```

```
<preamble>
```

```
\begin{document}
```

```
<doc body>
```

```
\end{document}
```

## Beispiel (für pdf $\LaTeX$ )

```
\documentclass{article}
% für pdf $\LaTeX$ 
\usepackage[T1]{fontenc}           % diese drei Pakete
\usepackage[ngerman]{babel}       % in dieser
\usepackage[utf8]{inputenc}       % Reihenfolge

\begin{document}

\section{Hallöchen}
Das ist nur eine mäßige Überraschung.

\end{document}
```

# Dokumententypen

- ▶ für kurze Dokumente
  - ▶ `article`, `scrartcl`, `amsart`
  - ▶ `llncs`, ...
  - ▶ `letter`
  - ▶ ...
- ▶ für lange Dokumente
  - ▶ `report`, `book`, `scrreprt`, `scrbook`
  - ▶ `memoir`
  - ▶ `tufte-book`
  - ▶ ...
- ▶ für spezielle Dokumente
  - ▶ Folien: `beamer`, ...
  - ▶ Poster: ...



# Präambel

- ▶ Laden zusätzlicher Pakete

`\usepackage{⟨Paket⟩}`

- ▶ Modifikationen von  $\LaTeX$
- ▶ Erweiterungen

- ▶ Festlegung von Einstellungen

- ▶ keine einheitliche Syntax
- ▶ z. B. `\usepackage[⟨Optionen⟩]{⟨Paket⟩}`

- ▶ eigene Definitionen

- ▶ für Textauszeichnungen

⋮

- ▶ wilde Hacks

# Apropos wilde Hacks

das folgende ist nicht  $\LaTeX$ , sondern plain  $\TeX$   
aber ansonsten ...

```
\let~\catcode~~76~~A13~~F1~~j00~~P2jdefA71F~~7113jdefPALLF
PA' 'FwPA;;FPAZZFLaLPA//71F71iPAHHFLPAzzFenPASSFthP;A$$FevP
A@@FfPARR717273F737271P;ADDFRgniPAW71FPATTFvePA**FstRsamP
AGGFRruoPAqq71.72.F717271PAY7172F727171PA??Fi*LmPA&&71jfi
Fjfi71PAVVFjbigskipRPWGAUU71727374 75,76Fjpar71727375Djifx
:76jelse&U76jfiPLAKK7172F7117271PAXX71FVLn0SeL71SLRyadR@oL
RrhC?yLRurtKFeLPFovPgaTLtReRomL;PABB71 72,73:Fjif.73.jelse
B73:jfiXF71PU71 72,73:PWs;AMM71F71diPAJJFRdriPAQQFRsreLPAI
I71Fo71dPA!!FRgiePBt'el@ lTLqdrYmu.Q.,Ke;vz vzLqip.Q.,tz;
;Lql.IrsZ.eap,qn.i. i.eLlMaesLdRcna,;!;h htLqm.MRasZ.il,k,%
s$;z zLqs'.ansZ.Ymi,/sx ;LYegseZRyal,@i;@ TLRlogdLrDsW,@;G
LcYlaDLbJsW,SWXJW ree @rzcLhzsW;;WERcesInW qt.'oL.Rtrul;e
doTsW,Wk;Rri@stW aHAHHFndZPpqar.tridgeLinZpe.LtYer.W,:jbye
```

# Dokumentenrumpf

das „eigentliche“ Dokument

# Dokumentenrumpf

das „eigentliche“ Dokument

- ▶ Inhalt
- ▶ Struktur
- ▶ äußere Form

# Dokumentenrumpf

das „eigentliche“ Dokument

- ▶ Inhalt
  - ▶ vom Autor
- ▶ Struktur
  - ▶ vom Autor
  
- ▶ äußere Form

# Dokumentenrumpf

das „eigentliche“ Dokument

- ▶ Inhalt
  - ▶ vom Autor
- ▶ Struktur
  - ▶ vom Autor
  - ▶ *Textauszeichnungen*, wie z. B.
    - ▶ `\section{Einführung}`
    - ▶ `\emph{wichtig}`
- ▶ äußere Form

# Dokumentenrumpf

das „eigentliche“ Dokument

- ▶ Inhalt
  - ▶ vom Autor
- ▶ Struktur
  - ▶ vom Autor
  - ▶ *Textauszeichnungen*, wie z. B.
    - ▶ `\section{Einführung}`
    - ▶ `\emph{wichtig}`
- ▶ äußere Form
  - ▶ teils von  $\LaTeX$
  - ▶ teils vom Autor !?

## Dokumentenrumpf (2)

- ▶ Die meisten Zeichen stehen für sich.
- ▶ übliche Ausnahmen
  - ▶ `\` `{` `}`
  - ▶ `%`
  - ▶ `$` `-` `^`
  - ▶ `&`
  - ▶ `~`
  - ▶ `#`
- ▶ `pdf $\LaTeX$` : für Umlaute etc. muss man „bitte“ sagen
  - ▶ `\usepackage[utf8]{inputenc}` oder
  - ▶ `\usepackage[latin1]{inputenc}`
- ▶ `lua $\LaTeX$` : erwartet Eingabe in UTF-8



## Dokumentenrumpf (3)

Im laufenden Text erzeugt man die „Sonderzeichen“ so:

- ▶ `\textbackslash` ergibt `\`
- ▶ `\^{}{}` ergibt `^`
- ▶ `\~{}{}` ergibt `~`
- ▶ ansonsten durch Voranstellen eines `\`  
z. B. `\{` ergibt `{`

# Überblick

Allgemeines

Prinzipieller Aufbau einer  $\LaTeX$ -Datei

„Feine Grobstruktur“ von  $\LaTeX$ -Dokumenten

`article` und ähnliches

`report` und ähnliches

Leerzeichen, Leerzeilen, Kommentare

## Die Struktur eines Dokuments ...

...hängt vom Dokumententyp (`\documentclass`) ab:

- ▶ mehr oder weniger ähnlich:
  - ▶ kürzer: Aufsätze (`article`, ...)
  - ▶ länger: Berichte (`report`, `classicthesis`, ...)
  - ▶ Bücher (`book`, `memoir`, ...)
- ▶ ziemlich anders
  - ▶ Folien (`beamer`, ...)
  - ▶ Poster (`beamerposter`, ...)
- ▶ ganz anders
  - ▶ Briefe (`letter`, ...)
  - ▶ Lebensläufe (`moderncv`, ...)

# Aufbau `article`-ähnlicher Dokumente

## Präambel

- ▶ `\author{<Autor(en)>}`
- ▶ `\title{<Titel>}`
- ▶ `\date{<Datum>}`

## Rumpf

- ▶ *Kopf*
  - ▶ `\maketitle`
- ▶ *Kurzfassung*
  - ▶ zwischen `\begin{abstract}` und `\end{abstract}`
- ▶ kein Inhaltsverzeichnis
- ▶ *Text* strukturiert in Abschnitte, Unterabschnitte, usw.
- ▶ *Literaturverzeichnis* (siehe späteres Kapitel)

# Abschnitte

`\section{<Überschrift>}`

- ▶ oberste Strukturebene bei `article`, ...
- ▶ Bedeutung:
  - ▶ markiert Beginn eines Abschnittes  
Ende: vor Beginn des nächsten bzw. `\end{document}`
  - ▶ legt Überschrift fest
- ▶ Effekte:
  - ▶ automatische Abschnittsnummerierung fortgeschaltet
  - ▶ setzt Nummer und Überschrift
  - ▶ Rücksetzen des Zählers für Unterabschnitte

## Unterabschnitte

`\subsection{<Überschrift>}`

- ▶ zweitoberste Strukturebene bei `article`
- ▶ Bedeutung:
  - ▶ markiert Beginn eines Unterabschnittes  
Ende: vor Beginn des nächsten Unterabschnittes, Abschnittes  
oder bzw. `\end{document}`
  - ▶ legt Überschrift fest
- ▶ Effekte:
  - ▶ automatische Nummerierung der Unterabschnitte  
fortgeschaltet
  - ▶ setzt Nummer und Überschrift

# Aufbau von `report`-ähnlichen Dokumenten

## Unterschiede zu `article`

- ▶ *Kapitel* „oberhalb“ von Abschnitten
  - ▶ `\chapter{\langle Überschrift \rangle}`
- ▶ *Inhaltsverzeichnis*
  - ▶ `\tableofcontents`
  - ▶ Default von  $\LaTeX$  nicht sehr schön
  - ▶ siehe späteres Kapitel

# Überblick

Allgemeines

Prinzipieller Aufbau einer  $\LaTeX$ -Datei

„Feine Grobstruktur“ von  $\LaTeX$ -Dokumenten

Leerzeichen, Leerzeilen, Kommentare



## Leerzeichen und Leerzeilen

- ▶ *Leerzeichen*: Zeichen mit den ASCII-Codes 32 («SPACE») und 9 («TAB»)
- ▶ Leerzeichen trennen Wörter
- ▶ mehrere unmittelbar hintereinander stehende Leerzeichen wirken wie ein einzelnes
- ▶ *Leerzeile*: eine Zeile, in der höchstens Leerzeichen (oder gar keine Zeichen) vorkommen
- ▶ Der nachfolgende Text beginnt in der Ausgabe einen neuen Absatz.

## Zeilenende

- ▶ verschiedene Methoden, um Zeilende zu markieren
  - ▶ Linux & Co, Mac OS X: Zeichen mit dem ASCII-Code 10 («LF»)
  - ▶ Windows: Zeichenfolge von ASCII-Codes 10 und 13 («CR»«LF»)
  - ▶ die umgekehrte Folge «LF»«CR» und nur «CR» kommen auch vor
- ▶  $\text{\TeX}$  sei Dank: einfach *das* Zeilenendezeichen
- ▶ *Zeilenendezeichen*
  - ▶ das nicht auskommentiert ist und
  - ▶ dem keine Leerzeile folgt
  - ▶ wirkt wie ein Leerzeichen

## Kommentare

- ▶ *Prozentzeichen %* markiert Anfang eines Kommentars
- ▶ Kommentar reicht immer bis einschließlich zum nächsten Zeilenendezeichen.
  
- ▶ Leerzeichen am Anfang einer Zeile, die auf eine Zeile mit Kommentar folgt, werden ignoriert.

## Beispiele

```
keine  
Überraschung
```

keine Überraschung

```
auch  das  ni%  
cht
```

auch das nicht

```
Und das erst.  
%  
Ganz klar.
```

Und das erst. Ganz klar.

## Tipp für Kollaboration/Versionsverwaltung

«Diffs» sind übersichtlicher, wenn

- ▶ nur wenige Eingabezeilen betroffen und
- ▶ diese kurz sind.

Also ...

## Tipp für Kollaboration/Versionsverwaltung

«Diffs» sind übersichtlicher, wenn

- ▶ nur wenige Eingabezeilen betroffen und
- ▶ diese kurz sind.

Also ...

- ▶ kurze Eingabezeilen
  - ▶ und keine langen (z. B. ganze Sätze in einer Eingabezeile)

## Tipp für Kollaboration/Versionsverwaltung

«Diffs» sind übersichtlicher, wenn

- ▶ nur wenige Eingabezeilen betroffen und
- ▶ diese kurz sind.

Also ...

- ▶ kurze Eingabezeilen
  - ▶ und keine langen (z. B. ganze Sätze in einer Eingabezeile)
- ▶ falls der Editor beim Einfügen einzelner Wörter ansonsten viel automatisch umbricht:
  - ▶ Jeder Satz beginnt in einer neuen Zeile.
  - ▶ Dazwischen eine Zeile der Form  
%